NATIONAL INSTITUTE OF TECHNOLOGY ROURKELA

Improved Authentication Mechanism Based On Elliptic Curve Cryptography

by

Ankita Jena

A thesis submitted in partial fulfillment for the
degree of Bachelor of Technology

under the guidance of
Prof. P.M. Khilar
Department of Computer Science & Engineering

May 2013

Certificate

This is to certify that the Thesis Report entitled IMPROVEMENTS IN ELLIPTIC CURVE CRYPTOGRAPHY submitted by Ankita Jena (109CS0023) of Computer Science and Engineering during May 2013 at National Institute of Technology, Rourkela is an authentic work performed by them under my supervision and guidance. To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other University / Institute for the award of any Degree or Diploma.

Signed:

(Dr. Pabitra M. Khilar
Department of Computer Science & Engineering
National Institute of Technology Rourkela)
Date:

Abstract

This project is based on the study of cryptographic methods implemented in public key cryptography. Initially, to gain an insight into the methods, both RSA cryptosystem and Elliptic Curve Cryptosystem has been implemented and their performance is compared on the basis of execution time. Due to the better performance of ECC, the project focuses on ECDSA i.e. Elliptic Curve Digital Signature Algorithm. It has been studied thoroughly and its weaknesses assessed. Then a new algorithm has been proposed using the origianal ECDSA with montgomery scalar multiplication in modified jacobian coordinates. The results in context of execution time have been compared and it is seen that the new method is more efficient.

Acknowledgements

Firstly, I wish to express my deep sense of gratitude to Prof P. M. Khilar, Dept. of Computer Science and Engineering, National Institute of Technology, Rourkela, my guide, for his consistent encouragement, incalculable guidance and co-operation to carry out this project, and for giving me an opportunity to work on this project and providing me with a great environment to carry my work in ease. I would also like to thank my friends, Ritika Ojha, Ankit Saroha and Anjana Tudu, for providing insightful comments and helping enhance the quality of thesis, and family, for being understanding and supportive throughout. Their help cannot be penned with words.

Contents

Chapter 1

Introduction

1.1 Introduction

Security is one of the most important parts of the modern computer science. RSA Cryptography and Elliptic Curve Cryptography are the public key cryptography systems which are used these days.

WHAT IS CRYPTOGRAPHY?

Cryptography has its meaning in the Greek words kryptos and graphos meaning hidden and written respectively. So cryptography is the science or art of converting text to a coded form that makes the text unreadable for those people you dont want to read it.

WHAT IS ENCRYPTION AND DECRYPTION?

The process of converting plain text to cipher text using some mechanism is called encryption. Decryption is converting the cipher text back to plain text form.

WHAT IS PRIVATE KEY CRYPTOGRAPHY?

In private key cryptography, the encryption and decryption both are done using the same key. Examples are AES and DES. In this Alice uses a key to encrypt the message and secretly shares the key with Bob who then uses it to decrypt the encrypted message.

WHAT IS PUBLIC KEY CRYPTOGRAPHY?

Public key cryptography is also known as asymmetric key cryptography. Bobs public key is known to everyone. So suppose Alice wants to send a message to Bob, she encrypts the message with Bobs public key and Bob decrypts it with his private key which is known just to him. This is slower but more secure. Examples are RSA, Diffie Hellman and ECC.

WHAT IS A KEY?

A key is basically a value that is sued in an algorithm for cryptography to convert plain text to cipher text. It has a huge value and is measured in bits. The bigger the key is

in a public key cryptography, the more secure is the cryptographic mechanism.

DIGITAL SIGNATURE

Digital signatures are one of the numerous benefits of Public key cryptography. They are employed in the encryption mechanism for validating identity of sender and receiver i.e. ensuring authentication. Thus the digital signatures also provide integrity. It helps in tackling non-repudiation that is that a sender cannot deny that he has not sent the message. Its function is same as that of a signature but there is one advantage, it cannot be forged. Thus it is way more secure. It is usually seen that people tend to use digital signature without encryption at times when they do not care about the information but the senders or receivers ID of the information.

Chapter Organisation In this chapter, in section 1.1 a brief introduction to the project has been given. In section 1.2, the motivation behind the work has been explained. In section 1.3 the objectives of the project have been mentioned. In section 1.4 an overall guide to the thesis has been presented and finally in 1.5 a befitting conclusion to the chapter has been given.

1.2 Motivation

The motivation behind studying cryptography and trying to being out improvements is the need to ensure the security of a given message. There are three main goals of security that always need to be taken care of and ensuring these was my motivation for taking up the project. They are confidentiality, Integrity and Availability.

1.2.1 Confidentiality

This property as the name suggests is about keeping information private and far from prying eyes. Only the people with the authority to access should be able to retrieve it. Information has been the source of power since time immemorial, and in this age of technology, it is ever so more important and keeping it private is of the utmost priority. So in order to preserve the confidentiality of information, the information is encrypted with only the authorized personnel being able to decrypt it because of some information known only to them. There are two main threats to confidentiality, snooping and traffic analysis.

1.2.2 Integrity

When a sender transmits message, ensuring that the receiver receives the message as it was, wholly and error free, also that there has been no modification and that the sender is the one who sent that message, come under the integrity section. It has two main mechanisms to ensure integrity and they are preventive mechanism and detective mechanism. Preventive mechanisms are the ones that do not let an attacker modify the information in any way where as the detective mechanisms detect if there have been any modifications.

1.2.3 Availability

Availability is the part that ensures that those who have the rights to the information have always got the access to it i.e. the information is available to them when needed. There is no use of confidentiality and integrity if the authorized users cannot get the information they are entitled to. It is one of the most important characteristics.

1.3 Objective

With ensuring the three main goals of security, confidentiality, integrity and availability, being my motivation, my objective is to study the different types of attacks possible on the information and tackle them with the right types of counter measures. Also optimize the process of countering by proposing new method. In short:-

1. Study and analyse the popular public key cryptosystems like RSA cryptographic method and Elliptic Curve Cryptography method and verify that ECC is more efficient than RSA on the basis of execution time

2. Study in detail the authentication mechanism in Elliptic Curve Cryptography i.e. Elliptic Curve Digital Signature Algorithm[5] and improve upon it by using Montgomery Scalar Multiplication[6] and Modified Jacobian co-ordinates[1].

1.4 Thesis Organisation

In the thesis of my project, up till now I have given you a brief introduction to the prerequisite knowledge necessary for my project. In the further sections I will be discussing the details. In chapter 2, I will discuss the research done regarding this project, with

all the papers studied and implementations carried out as a prequel to the main work. In chapter 3, I have gone through the area of the subject I mainly focused on and what improvements I made to them. I have discussed the system model, the analysis and the results of my simulation. In chapter 4, I have given a brief look into the future work possible. In chapter 5, I have concluded my thesis work.

1.5 Conclusion

In this chapter, I discussed all the matter that needed to be known to be introduced into the world of Cryptography which vast and a vat of intellectual property. Knowing the basics, one can now delve deeper into the subject matter of the project. In short, this chapter provided the meaning of cryptography and the concepts of public key and digital signature. Also, the reason behind the project i.e. motivation and the objective of the project.

Chapter 2

Related Work

2.1 Introduction

In this section, I discuss two of the most popular cryptosystems: Elliptic Curve Cryptography and Rivest Shamir Adleman Cryptography. A lot of papers were studied about both the cryptosystems that are listed in section 2.2 and the algorithms of ECC and RSA were implemented in Java language and their efficiencies in context of execution time were compared. In section 2.3 and 2.4 ECC and RSA have been discussed respectively.

Chapter Organisation In this chapter the different sections discussed are- in section 2.1 we give an itroduction to the related work carried out. In section 2.2 we go through the papers in brief that have been studied. In section 2.3 and 2.4 ECC and RSA have been discussed in details respectively. In 2.5 I have presented the results and in 2.6, the conclusion has been given.

2.2 Literature Survey

Implementing Digital Signature with RSA Encryption Algorithm to Enhance the Data Security of Cloud in Cloud Computing- Uma Somani, Kanika Lakhani, Manish Mundra [9]

Abstract : Even if virtualization and cloud computing help a company to achieve a lot, it will be truly beneficial if it ensures security. In order to utilize the cloud computing, one must share or transfer data and thus comes the issue to preserve the confidentiality, integrity and availability of data. To achieve this, the paper discusses RSA cryptography. An algorithm is proposed for key generation, signature verification and data encryption and decryption. Using this, the CIA of data is successfully preserved.

Data Security in Cloud Computing with Elliptic Curve Cryptography [10]

Abstract : This paper discusses the emerging trend of the cloud computing and the growing need for security in it. It discusses the various security risks faced by the cloud like regulatory compliance, privileged user access, data location, data t segregation, recovery, investigative support and long term viability. It then discusses the proposed security mechanisms like encryption, authentication and access control. It discusses the application of Elliptic curve cryptography to achieve these and its working principle.

Fast Multiplication on Elliptic Curves over GF(2m) without Pre-computation- Julio Lpez and Ricardo Dahab[3]

Abstract : In this paper, algorithms for carrying out multiplications on elliptic curves in a faster way have been discusses, these elliptic curves are non-singular in nature and defined over GF(2m). The existing method, i.e. Montgomery method has been studied and optimized for faster implementation in both software and hardware. It focuses on relating the improvement of the proposed algorithm over the older addition subtraction method of multiplication and also removing any need for pre-computation making it possible to implement on places with less memory. It has also discussed the advantages of the improved method.

Software Implementation of Elliptic Curve Cryptography over Binary Fields - Darrel Hankerson, Julio Lpez Hernandez, and Alfred Menezes [11]

Abstract : In this paper, the main focus is on studying diligently and exhaustively the software application of the elliptic curves recommended by NIST on various workstations. ECC was developed by Neal Koblirz and Victor Miller independently and has become a very popular public key cryptography which a lot of research being done on them. So here different mechanism to improve upon the ECC have been proposed as it has become acknowledged by standards like ANSI, IEEE, ISO and NIST.

The Elliptic Curve Digital Signature Algorithm ECDSA- Don Johnson and Alfred Menezes[5]

Abstract : ECDSA stands for Elliptic Curve Digital Signature Algorithm. It is the implementation of digital signature with the help of elliptic curves. It is an accepted standard by ANSI, NIST, ISO and IEEE. This paper discusses the various properties of elliptic curves, the concept behind the operations carried out on elliptic curves, the algorithm proposed for digital signature over the elliptic curves and also the discrete logarithmic problem associated with it.

2.3 Elliptic Curve Cryptography

2.3.1 INTRODCUTION TO ECC

Proposed in 1985 by Neal Koblitz and Victor Miller independently, Elliptic Curve Cryptography has become one of the most popular public key cryptography till date. It is highly secure owing to its exponential nature of discrete logarithmic problem. It is based on an elliptic curve, that has the basic equation[6]

$$y^2 = x^3 + ax + b$$

2.3.2 DISCRETE LOGARITHMIC PROBLEM OF ECC

Let us takes two points P and Q such that they lie on the elliptic curve being considered. Suppose it satisfies the equation Q = kP, then determining k with Q and P known is known as the elliptic curve discrete logarithmic problem. It is basically the measure of the infeasibility of calculation of k using computational resources. In order to make ECDLP completely intractable, we have to select the right elliptic curve and a q such that the number of points on the elliptic curve over the prime q is divisible by a large prime or q is a huge prime number on its own[1].

2.3.3 ALGORITHM FOR ECC

There has to be some information that is publicly known to all the users, thus making it the public key cryptography. The publicly known entities are:-

1. From the equation of the elliptic curve, we need to know:-
   - the values of the constants a and b.
   - the value of m, where elliptic curve is defined over GF(2^m).
2. The group of the elliptic curve.
3. A base point B, i.e. any point on the curve E that belongs to the group taken as a base.

The algorithms for different parts of ECC are:-

2.3.3.1 Key Generation Algorithm

- Randomly select an integer *Apriv*. It acts as the private key for A.
- Then generate *Apub* such that **Apub = Apriv * B**, where Apub is the public key for A.
- Randomly select an integer *Bpriv*. It acts as the private key for B.
- Then generate *Bpub* such that **Bpub = Bpriv * B**, where Bpub is the public key for B.
- Finally, A generates key, **Ka = Apriv * Bpub**
- B generates key, **Kb = Bpriv * Apub**

2.3.3.2 Signature Generation Algorithm

- Calculation of message digest with a HASH function, preferable SHA-1, where e is the message digest, m is the message such that e = **HASHfun(m)**
- Generate a random integer *rand* between 1 and n-1.
- The first of the signature, *sign1* is calculated from **sign1 = x mod n** where x is the product of B with *rand* i.e. x = xcod(rand * B) where xcod is a function to get the x co-ordinate.
- But if *sign1* is 0, then redo the previous step.
- The second part of the signature, sign2 is calculated from the equation **sign2 = rand -1(e + (Apriv*sign1)(mod n)**
- But if sing2 is 0, then re-generate r and follow the procedure again.
- The signature generated is a pair **(sign1,sign2)**.

2.3.3.3 Signature Validation Algorithm

- Check if sign1 and sign2 lie between the range of 1 and n-1. If not, the signature is not valid.
- Calculate the message digest from the received message with the same hash function, **e = HASHfun(m)**.
- Calculate *var1*, where **var1 = sign2 1(mod n)**

- Calculate *var2*, such that **var2 = (e*var1) modn**
- Calculate *var3*, such that **var3 = (sign1*var1)** modn
- We then calculate X, such that **X = (var2*B) + (var3*Apub)**
- If **sign1(mod n)** is equal to **xcod(X)**, then signature is verified.

2.3.3.4 Encryption Algorithm

- The plain text M is mapped onto the elliptic curve at a point P.
- Generate a random integer *rand* between 1 and n-1.
- The cipher text is then encoded as a pair C, where **C = [(rand * B),(P + (rand * Bpub)]**

2.3.3.5 Decryption Algorithm

- Get x, where **x = xcod(C)**.
- Calculate *prod*, where **prod = Bpriv * x**
- Calculate **(P + (rand * Bpub)) prod)**, this gives the mapped point P
- Then un-map P to the plain text M

2.4 RSA CRYPTOGRAPHY

2.4.1 RSA CRYPTOGRAPHY

RSA was invented by Ron Rivest, Adi Shamir, and Leonard Adleman in 1978. RSA is based on the use of two exponents, e and d where e is public and d is private. Alice uses e to create cipher text from plain text and Bob uses d to decrypt it. It is the first public key cryptography that included the digital signature scheme in it[2].

2.4.2 DISCRETE LOGARITHMIC PROBLEM OF RSA

The main basis of the security of RSA lies in its discrete logarithmic problem. The discrete logarithmic problem here is based on n, the modulus that is used in the encryption algorithm. n is the product of the two large primes p and q, and with n known, the

problem is to factorize it to get p and q. Factorizing n is infeasible in polynomial time complexity and this ensures the security. At present RSA requires 1024 bit key, but starting 31st December ,2013, it will require 2048 bits.[8]

2.4.3 ALGORITHM FOR RSA

2.4.3.1 Key Generation Algorithm

- You have to choose two large primes prime1 and prime2 which are unique.
- Compute the modulus n by the equation **n=prime1 x prime2**
- Then the totient is to be calculated as phi(n), where **phi(n) = (prime1-1) x (prime-1)**
- A random integer pub is to be chosen so that pub and phi(n) are co-primes and e lies between 1 and phi(n)
- Another integer d is to be chosen such that it satisfies the equation **priv x pub =1 (mod phi(n))**
- (pub,n) is published as the public key and d is kept secret as the private key.

2.4.3.2 Signature Generation Algorithm

- Using a hash function, preferably SHA-1, create the message digest from M.
- Then the message digest is represented as an integer between 1 and n-1, msg.
- The signature is computed by the equation **sign = msg ^ priv (mod n)**.
- Send the signature to the recipient

2.4.3.3 Signature Validation Algorithm

- Use the public key pub, along with n, to calculate an integer ver,
  ver = sign^pub (mod n)
- Then get the message digest Mver from this
- If Mver is same as M, then the signature is verified.

2.4.3.4 Encryption Algorithm

- The message is represented as msg.
- The cipher text is calculated as C, where **C = msg ^ pub (mod n)**
- C is then sent

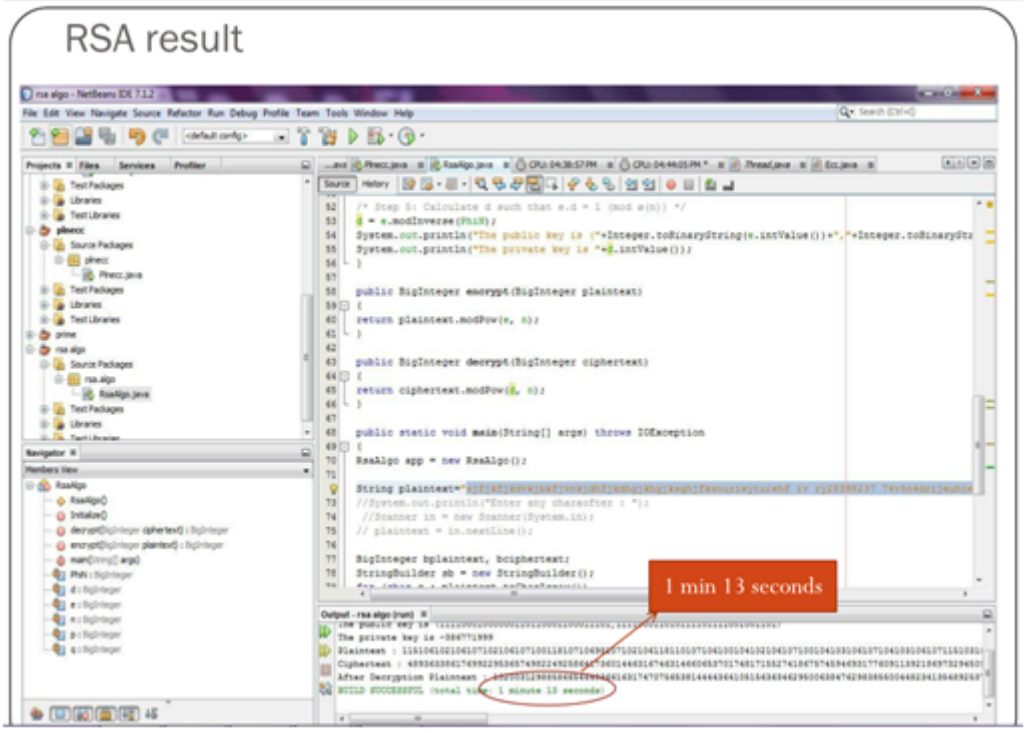
2.4.3.5 Decryption Algorithm

- The received message is represented as rmsg.
- The plain text is calculated as P, where **P = rmsg ^ pub (mod n)**
- P, the plain text is then retrieved.

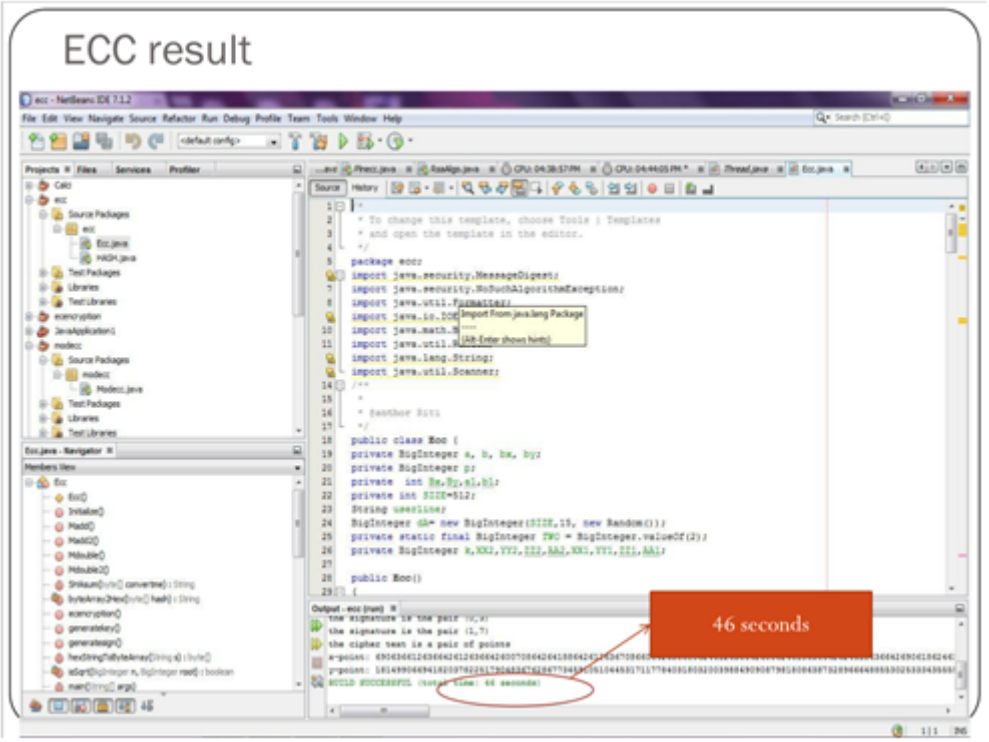
2.5 Results

Claim 1: Elliptic Curve Cryptography provides better security because of its exponential nature of discrete logarithmic problem (studied)
Claim 2: It runs faster than RSA (Verified by implementation in NetBeans)

RSA EXECUTION TIME



ECC EXECUTION TIME



Claim 3: As ECC is more efficient, I proposed to improve upon the existing ECDSA.

2.6 Conclusion

It was thus seen that ECC is more efficient than RSA. It has been studied that a 1024 bit key in case of RSA provides the same amount of security as a 163 bit key ECC [7]. Also we see ECC is faster than RSA. Hence ECC was selected to be improved upon.

Chapter 3

Proposed Algorithm

3.1 Introduction

In this section, we go through the Elliptic Curve Digital Signature Algorithm in detail. I studied the algorithm used for ECDSA and picked out the scope for changes. It is seen that it can have some weaknesses like being less secure as it can be easily attacked by side-channel analysis and also it is slower due to a large number of field inversions. Thus, I apply the changes to the existing algorithm using Montgomery multiplication and concept of projective co-ordinates, and compare the efficiency of proposed algorithm with the one proposed by Don Johnson and Alfred Menezes.

Chapter Organisation In this chapter in sction 3.1 a brief introduction to the work done is given. Then ECDSA is discussed in detail and the operation in it as well in section 3.2 along with the proposed algorithm. In the section 3.3, the algorithm proposed is gice. In section 3.4. the results have been shown and in section 3.5 a brief description of system model and the inputs are given. In section 3.6, the chapter has been concluded.

3.2 ECDSA

3.2.1 Overview

As stated earlier, the security of ECDSA depends upon the discrete logarithmic problem. And in the discrete logarithmic problem, the main part is

$$\mathbf{Q = k * P}$$

It is known as the point multiplication and this is the centre of ECDSA and hence, we will be improving upon the point multiplication.[5]

3.2.2 Point Multiplication

Point multiplication is basically made of two operations, point addition and point doubling.

Point Addition

Point addition is when we add two distinct points on the elliptic curve to get a third point, which by virtue of its group property, lies on the curve too. If P and Q are the two points, then their addition is say R such that when a line that joins P and Q is drawn, it connects the curve at some point say R , then the reflection of R on the x-axis gives us R. If Q is P, i.e. when P and Q are added, it never intersects the curve, then it is said to be inverse of P and there sum is O, point at infinity.

Point Doubling

Point doubling is basically when the two points we are adding, P and Q, are the same. In this case, a tangent to the curve is drawn through the point P and wherever it intersects the curve is marked as R. Now the resultant of the point doubling is R which is the reflection of R along x-axis.

The formula for point addition and doubling are:-

$$x_3 = \begin{cases} (\frac{y_1+y_2}{x_1+x_2})^2 + \frac{y_1+y_2}{x_1+x_2} + x_1 + x_2 + a \ , P \neq Q \\ x_1^2 + \frac{b}{x_1^2} \ , \qquad\qquad P = Q. \end{cases}$$

$$y_3 = \begin{cases} (\frac{y_1+y_2}{x_1+x_2})(x_1+x_3) + x_3 + y_1 \ , P \neq Q \\ x_1^2 + (x_1 + \frac{y_1}{x_1})x_3 + x_3 \ , \qquad P = Q. \end{cases}$$

Where the two points added are P(x1,y1) and Q(x2,y2) and the resultant R is (x3,y3). So, in point multiplication when we say Q= 7P, we mean

$$\mathbf{Q = (\ 2\ (\ (\ 2P\ ) + P\ ) + P\ )}$$

Now, in normal ECC, the point multiplication is done by a method called Double and Add method. But in our proposed algorithm, we follow Montgomery Scalar Multiplication method.

3.2.3 Double and Add Method[3]

3.2.3.1 Algorithm

If we represent k in the binary form, let it be (kl-1,kl-2,.k0)2
Input: kbin, Point1
Output: Point2
Pseudocode :-

1. Set Point2 as 0 and set a temporary variable Tpoint as point1.
2. Repeat steps 3 to 4 till i is greater than l-1
3. If value of kbin is 1 then do
   point2 = point2 + Tpoint
4. Set Tpoint as **2*Tpoint**
5. Return point2

3.2.3.2 Disadvantages

- Slow - This method is slow as it has involves too many field inversions which are as costly as 7 multiplications
- Not Secure - It can be easily attacked by side-channel attack. We see that when the bit value is 1, it is doing an additional operation of addition. So if there is some sort of operation flow leakage, then the attacker can easily tell the difference and can get significant amount of information.

3.2.4 OVERCOMING LESS SECURITY: MONTGOMERY SCALAR MULTIPLICATION

It was seen that the x-coordinate of the resultant point can be calculated on the basis of x-coordinates of P and Q[5]. This way new formula was derived which was then implemented in the algorithm.

$$x_3 = \begin{cases} x + (\frac{x_1}{x_1 + x_2})^2 + \frac{x_1}{x_1 + x_2} & , P_1 \neq P_2 \\ x_1^2 + \frac{b}{x_1^2} & , \quad P_1 = P_2. \end{cases}$$

$$y_1 = (x_1 + x)\{(x_1 + x)(x_2 + x) + x^2 + y\}/x + y$$

3.2.5 Algorithm

1. It converts the decimal value of k into the binary format kbin such that **k=** **($\mathbf{k}_{l-1}$,$\mathbf{k}_{l-2}$.$\mathbf{k}_1$,$\mathbf{k}_0$)**
2. Then we set tpoint as **tpoint = point1**
3. Set **point2 = point2 + b/point2**, where b is one of the elliptic curve constants
4. We repeats steps 5 through 8 till i is greater than l-1
5. Initialize a temporary variable temp, and set it
   temp=tpoint/(tpoint+point2)
6. If value of kbin is 1, then we do
   tpoint = point1 + temp^2 + temp, point1^2 = point2^2 + b/point2^2
   Else
   tpoint = point2 + temp^2 + temp, point2^2 = point1^2 + b/point1^2
7. Then we take two variables rand1 rand2 where
   rand 1 = tpoint + point1
   rand2 = point2 + point1
8. Set x-coordinate of the point x as tpoint
9. Set y-coordinate of the point y as
   rand1((rand1 x rand2) + point2 + point1.y) / (point1.x + point1.y)
10. Return (x,y).

3.2.5.1 Improvements

Now from the above algorithm we see that if the multiplicand is 1, then number of

- multiplications =1
- number of additions=3
- number of field inversions=2
- number of squaring=2

If the multiplicand is 0, the same numbers of operations take place, thus, making it safe from side-channel attack.

3.2.6 OVERCOMING SPEED: MODIFIED JACOBIAN CO-ORDINATES

The speed of the algorithm is hindered because of the numerous field inversions. Replacing field inversions with multiplications will improve the speed. Thus, comes the concept of projective co-ordinates. In this project, Modified Jacobian co-ordinates[4] have been used. The corresponding affine co-ordinates are represented as (X/Z^2, Y/Z^3) and the projective co-ordinates are (X, Y, Z, az^4). So, the operations in this co-ordinate are as follows:-

3.2.6.1 Point Addition

- $X_3= (H^3) - (2U_1H^2)+ R^2$
- $Y_3=-S_1H^3+R\ (U_IH^2-X_3)$
- $Z_3=Z_1Z_2H$
- $A_3=AZ_3{}^4$
- $U_1=X_1Z_2{}^2$
- $U_2=X_2Z_1{}^2$
- $S_1=Y_IZ_2{}^3$
- $S_2=Y_2Z_1{}^3$
- $H=U_2-U_1$
- $R=S_2-S_1$

3.2.6.2 Point Doubling

- X_3=T
- Y_3=M(S-T)-U
- Z_3=$Y_I Z_I$
- A_3=2UA_1
- S=4$X_1 {Y_1}^2$
- U=8${Y_1}^4$
- M=3${X_1}^2$+A_1
- T=-2S+M^2

3.3 Proposed Authentication Algorithm

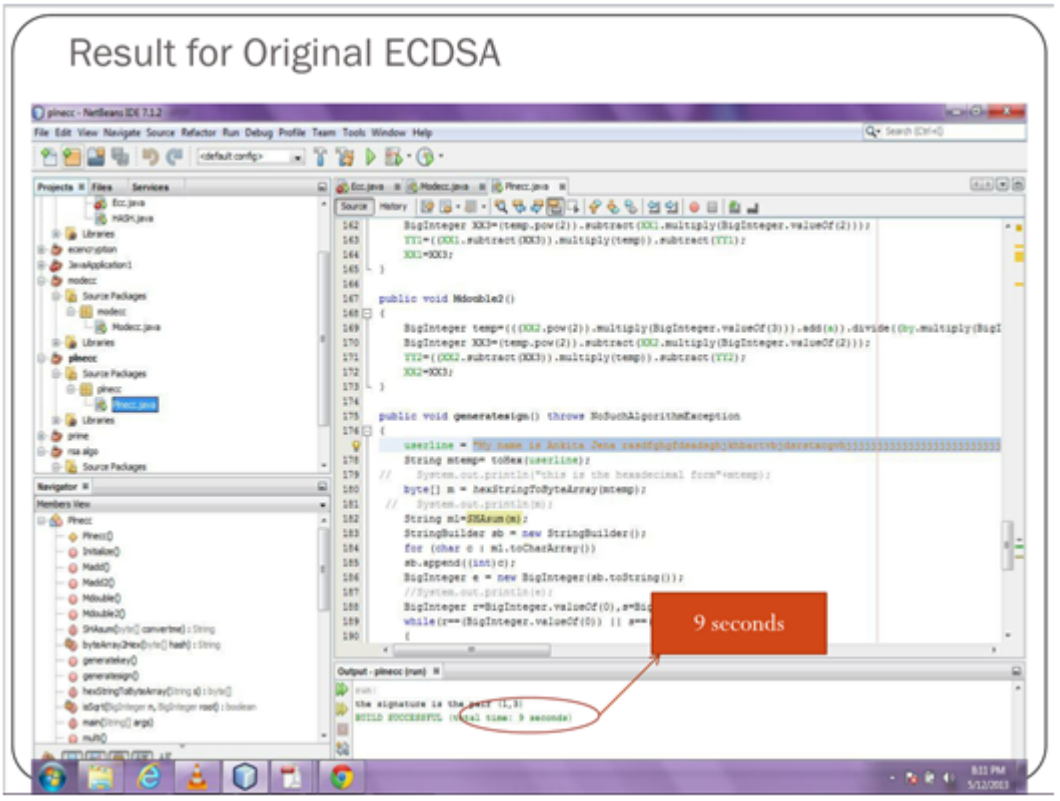
In this project it has been proposed that the Elliptic Curve cryptography be implemented using Montgomery Scalar multiplication for point multiplication in modified Jacobian coordinates.
The proposed algorithm is:

1. It converts the decimal value of k into the binary format kbin such that
   k= ($\mathbf{k}_{l-1}$,$\mathbf{k}_{l-2}$.$\mathbf{k}_1$,$\mathbf{k}_0$)
2. Then we set tpoint as **tpoint = point1**
3. Set **point2 = point2 + b/point2**, where b is one of the elliptic curve constants
4. We repeats steps 5 i is greater than l-2
5. If value of kbin is 1, then we do
   Madd(X1,Y1,Z1,A1,X2,Y2,Z2,A2), Mdouble(X2,Y2,Z2,A2)
   Else
   Madd(X2,Y2,Z2,A2,X1,Y1,Z1,A1), Mdouble(X1,Y1,Z1,A1)
6. **Return Mxy(X1,Y1,Z1,A1)**

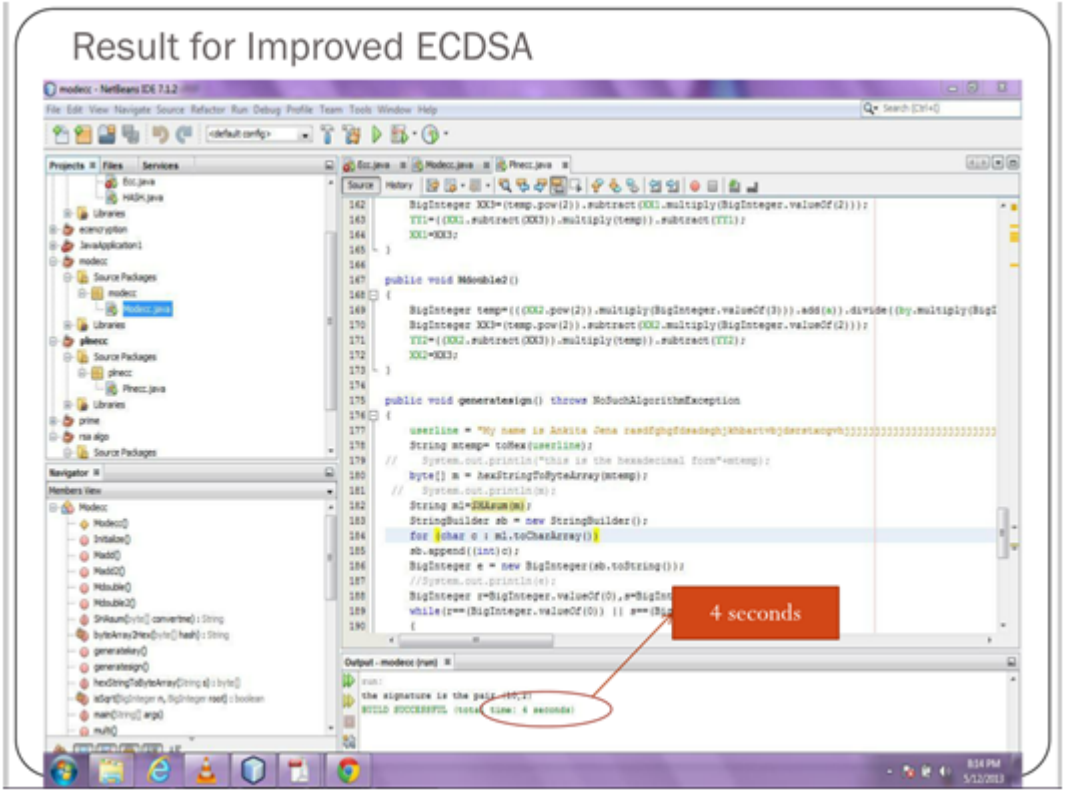
3.4 Analysis and Result

The original ECDSA was compared with the proposed ECDSA, and the following results were obtained.

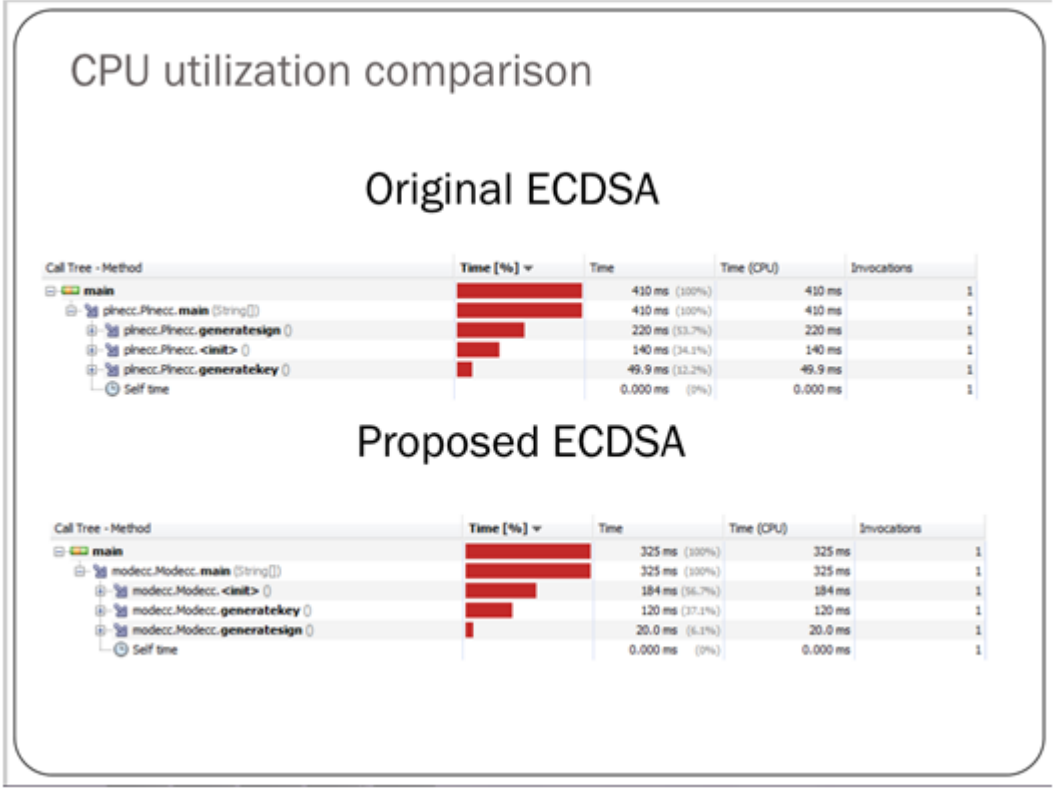
ORIGINAL ECDSA



IMPROVED ECDSA



CPU PERFORMANCE ANALYSIS: ORIGINAL(1) vs IMPROVED(2)



3.5 Inputs and System Model

The inputs given to both original and improves ECDSA are:-

- a=15
- b=6
- Base point B=(10,34)
- p=13
- The programs were implemented in NetBeans: NetBeans IDE 7.1.2 on Windows 7 system

3.6 Conclusion

Proposed algorithm is more efficient on the context of execution time than original ECDSA. The per cent of improvement is 55.55% It is seen that when ECDSA is implemented in Modified Jacobian co-ordinates with point multiplication done using Montgomery Scalar Multiplication it is indeed much more efficient than the one proposed by Don Johnson and Alfred Menezes[5].

Chapter 4

Conclusion and Future Work

4.1 Final Conclusion

- Studied and implemented RSA and ECC to compare the efficiency in terms of execution time
- Improved on ECC by implementing ECDSA using Montgomery Scalar Multiplication for point multiplication in modified jacobian coordinates and got faster execution than original ECDSA

4.2 Future Work

- Claim 1 : Can be implemented in Clouds
- Claim 2: the multiplication with 'b' in the Montgomery scalar multiplication can be optimized

Bibliography

1. Pritam Gajkumar Shah- Investigating Effects of Co-ordinate System on Execution Time of Elliptical Curve Protocol in Wireless Sensor Networks, 2012, 2012 International Conference on Future Communication Networks (IEEE)

2. Zhaohui Cheng and Manos Nistazakis- Implementing Pairing-Based Cryptosystems, 3rd International Workshop In Wireless Security Technologies

3. Julio Lopez and Ricardo Dahab- Fast Multiplication on Elliptic Curves over GF(2m) without Precomputation, Springer

4. Yiliang Han and XiaoyuanYang-Authenticated Public-key Encryption Based on Elliptic Curve, doi.ieeecomputersociety.org/10.1109/ICESS.2005.34

5. The Elliptic Curve Digital Signature Algorithm ECDSA- Don Johnson and Alfred Menezes, 2000, Springer

6. Cetin K. Koc and TolgaAcar- Montgomery Multiplication in GF(2k), April 1998, Electrical Computer Engineering Oregon State University, Corvallis, Oregon 97331

7. Matthieu Rivain- Fast and Regular Algorithms for Scalar Multiplication over Elliptic Curves

8. Nicholas Jansma and Brandon Arrendondo- Performance Comparison of Elliptic Curve and RSA Digital Signatures, April 28, 2004

9. Implementing Digital Signature with RSA Encryption Algorithm to Enhance the Data Security of Cloud in Cloud Computing- Uma Somani, Kanika Lakhani, Manish Mundra

10. Data Security in Cloud Computing with Elliptic Curve Cryptography by Veerraju Gampala, Srilakshmi Inuganti, Satish Muppidi

11. Software Implementation of Elliptic Curve Cryptography over Binary Fields- Darrel Hankerson, Julio Lpez Hernandez, and Alfred Menezes